AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method of predicting cache performance, comprising:

storing data references applied to an operational data cache in a data processing environment;

applying said data references to a cache simulator configured to simultaneously simulate a plurality of caches of different sizes, wherein said cache simulator comprises multiple segments and each said simulated cache comprises one or more of said segments; and

generating for each of said plurality of simulated caches an estimate of performance based on said simulation, wherein said generating comprises:

for a first simulated cache in said plurality of simulated caches:

calculating the number of hits in all segments of said cache simulator that do not comprise part of said first simulated cache,

adding the number of said misses of all of said segments to said calculated number of hits to produce an initial estimated miss rate for said first simulated cache,

calculating a correction factor to apply to said initial estimated miss rate for said first simulated cache, and

multiplying said initial estimated miss rate for said first simulated cache by said correction factor to yield a predicted miss rate for said first simulated cache;

wherein each application of one of said data references to said cache simulator causes either a hit in one of said segments or a miss of every said segment.

2. (Original) The method of claim 1, further comprising:

dynamically adjusting the size of said operational data cache to match the size of one of said simulated caches.

3. (Original) The method of claim 1, wherein said data processing environment is a database management system and said operational data cache is a buffer cache configured to cache data in said database management system.

4. (Original) The method of claim 3, wherein said storing data references comprises:

receiving a first data reference at said operational data cache;

storing said first data reference in a trace buffer for use in said simulation of said plurality of caches if said trace buffer is not full; and

discarding said first data reference if said trace buffer is full.

5. (Original) The method of claim 1, wherein said cache simulator comprises a list of simulated buffers, and each said simulated buffer is configured to store:

an identifier of a data item; and

an identifier of said segment of said cache simulator in which said simulated buffer is located.

6. (Original) The method of claim 5, wherein said applying comprises:

retrieving a first stored data reference;

identifying a first data item referenced in said first data reference; and

searching said cache simulator for a buffer in which an identifier of said first data item is stored.

7. (Original) The method of claim 6, further comprising:

if a first buffer is found in said cache simulator that stores an identifier of said first data item:

incrementing a hit counter for said segment in which said first buffer is found;

moving said first buffer to a head of said cache simulator; and

updating said stored segment identifiers of said buffers as necessary.

8. (Original) The method of claim 6, further comprising:

if no buffer is found in said cache simulator that stores an identifier of said first data item:

incrementing an absolute miss counter;

storing an identifier in a first buffer in said cache simulator;

moving said first buffer to a head of said cache simulator; and

updating said stored segment identifiers of said buffers as necessary.

9-10 (Canceled).

11. (Currently amended) The method of ~~claim 10~~claim 1, wherein said correction factor comprises the ratio of misses incurred during application of said

data references to said operational data cache to an initial estimated miss rate for a second simulated cache in said plurality of simulated caches;

wherein the number of buffers in said second simulated cache matches the number of buffers in said operational data cache.

12 (Canceled).

13. (Original) A method of simulating the performance of multiple caches of different sizes, comprising:

storing references to data items received during operation of a database management system;

maintaining a multi-segmented cache simulator comprising simulated buffers configured to store data identifiers, wherein each of said multiple caches comprises a set of said segments different from the other caches;

for each of said stored references:

searching said cache simulator for a first simulated buffer storing an identifier of said referenced data item;

if said first simulated buffer is found in said segmented memory:

incrementing a hit counter for said segment in which said first simulated buffer is located; and

moving said simulated cached reference to the head of said segmented memory;

if no simulated buffer storing an identifier of said referenced data item is found in said cache simulator:

incrementing a miss counter;

storing an identifier of said data item in a second simulated buffer; and storing said second simulated buffer at a head of said cache simulator; and

generating an estimate of the performance of each of said multiple caches.

14. (Original) The method of claim 13, wherein said estimated performance comprises a number of misses, wherein said generating comprises:

for a first cache of said multiple caches, calculating the number of said referenced data items for which identifiers were not found in the cache simulator segments comprising said first cache.

15. (Original) The method of claim 14, further comprising:

calculating a predicted number of misses for said first cache by multiplying said estimated performance for said first cache by a correction factor.

16. (Original) The method of claim 15, wherein:

said data item references are received at an operational cache of said database management system and result in a known number of misses in said operational cache;

a second cache of said multiple caches comprises a number of simulated buffers equivalent to the number of buffers comprising said operational cache; and

said correction factor is equal to the ratio of said known number of misses to said estimated performance for said second cache.

17. (Original) The method of claim 16, further comprising:

dynamically altering the size of said operational cache to the size of said first cache, wherein said predicted number of misses for said first cache is less than said known misses in said operational cache.

18. (Original) The method of claim 13, wherein said storing comprises:

receiving a first reference to a data item at said database management system;

if a trace memory configured to store said first reference is full, discarding said first reference; and

otherwise, storing said first reference in said trace memory without locking said trace memory.

19. (Original) The method of claim 13, wherein said data item references are received at an operational cache of said database management system, further comprising:

identifying the number of operational cache misses resulting from application of said data item references to said operational cache; and

dynamically altering the size of said operational cache to the size of one of said multiple caches having a superior estimated performance.

20. (Original) A computer readable storage medium storing instructions that, when executed by a computer, cause the computer to perform a method of predicting cache performance, the method comprising:

storing data references applied to an operational data cache in a data processing environment;

applying said data references to a cache simulator configured to simultaneously simulate a plurality of caches of different sizes, wherein said cache simulator comprises multiple segments and each said simulated cache comprises one or more of said segments; and

generating for each of said plurality of simulated caches an estimate of performance based on said simulation, wherein said generating comprises:

for a first simulated cache in said plurality of simulated caches:

calculating the number of hits in all segments of said cache simulator that do not comprise part of said first simulated cache,

adding the number of said misses of all of said segments to said calculated number of hits to produce an initial estimated miss rate for said first simulated cache,

calculating a correction factor to apply to said initial estimated miss rate for said first simulated cache, and

multiplying said initial estimated miss rate for said first simulated cache by said correction factor to yield a predicted miss rate for said first simulated cache;

wherein each application of one of said data references to said cache simulator causes either a hit in one of said segments or a miss of every said segment.

21-27 (Canceled).